

Structure-oriented Networks of Shape Collections

Noa Fish¹ Oliver van Kaick² Amit Bermano³ Daniel Cohen-Or¹

¹Tel Aviv University ²Carleton University ³Princeton University

Abstract

We introduce a co-analysis technique designed for correspondence inference within large shape collections. Such collections are naturally rich in variation, adding ambiguity to the notoriously difficult problem of correspondence computation. We leverage the robustness of correspondences between similar shapes to address the difficulties associated with this problem. In our approach, pairs of similar shapes are extracted from the collection, analyzed and matched in an efficient and reliable manner, culminating in the construction of a network of correspondences that connects the entire collection. The correspondence between any pair of shapes then amounts to a simple propagation along the minimax path between the two shapes in the network. At the heart of our approach is the introduction of a robust, structure-oriented shape matching method. Leveraging the idea of projective analysis, we partition 2D projections of a shape to obtain a set of 1D ordered regions, which are both simple and efficient to match. We lift the matched projections back to the 3D domain to obtain a pairwise shape correspondence. The emphasis given to structural compatibility is a central tool in estimating the reliability and completeness of a computed correspondence, uncovering any non-negligible semantic discrepancies that may exist between shapes. These detected differences are a deciding factor in the establishment of a network aiming to capture local similarities. We demonstrate that the combination of the presented observations into a co-analysis method allows us to establish reliable correspondences among shapes within large collections.

Keywords: Shape collections, similarity, correspondence, segmentation transfer

Concepts: •Computing methodologies → Shape analysis;

1 Introduction

Establishing a correspondence between two shapes is a well-researched and fundamental problem with many applications in various domains [van Kaick et al. 2011b]. Recently, special focus has been given to the computation of correspondences among shapes in a collection [Huang et al. 2012; Kim et al. 2013; Huang et al. 2014]. However, in the general case, computing a correspondence is an ill-posed problem as there is no well-defined set of rules to establish a mapping between two shapes. On the other hand, shape correspondence is highly intertwined with the problem of estimating the similarity between two shapes. A correspondence between two shapes is less ambiguous and inferred more easily when the

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or to publish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org. © 2016 ACM.

SA '16 Technical Papers, December 05-08, 2016, Macao.

ISBN: 978-1-4503-4514-9/16/12

DOI: <http://dx.doi.org/10.1145/2980179.2982409>

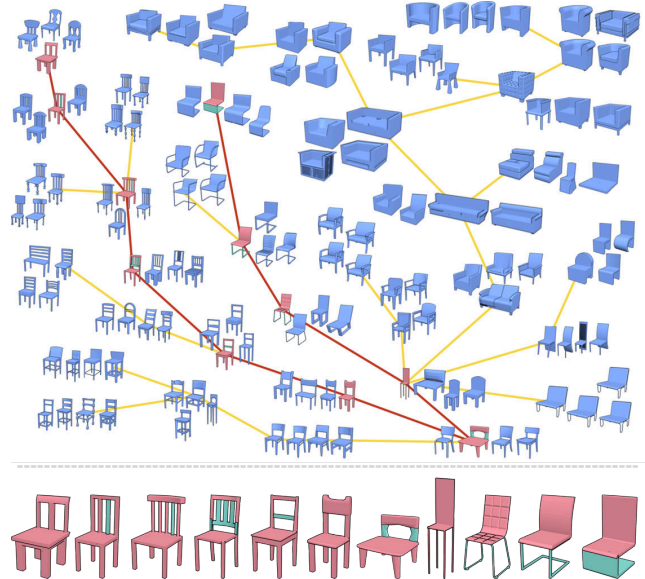


Figure 1: A structure-oriented network of a collection of shapes. By building a network of correspondences with an emphasis on structural differences among the shapes, we derive a network of clusters of shapes with near-equivalent structure, such that edges capture a change of structure between two connected clusters. We highlight a path traversed in the representation with red edges, and display the corresponding sequence of shapes at the bottom. Structural changes between successive shapes along the path are marked as green segments, and pertain either to the preceding or the following shape.

shapes are similar. Thus, our premise in this work is that the problem of computing correspondences in a collection can profoundly gain from considering the similarity between the shapes.

A good similarity measure should generally correlate with human perception and intuition. We observe that it is easy to identify when two shapes are identical or closely similar, but it is unclear how to account for large differences and effectively quantify them. Geometric differences between shapes can vary in magnitude in a continuous manner, so it is difficult to determine exactly at what point two shapes become dissimilar enough. In contrast, structural differences are discrete and therefore more pronounced. As such, they are more tangible and easier to identify, and even a small structural difference can serve as an immediate cue indicating a large dissimilarity between two shapes. We follow these observations to mitigate the problem of correspondence computation, in an effort to provide more robust results.

In this paper, we present a structure-oriented co-analysis technique that enables the creation of a network of correspondences within a large shape collection. By aiming to establish correspondences across an entire collection of shapes in a robust manner, we are faced with the difficulty of determining correspondences between dissimilar shapes. To alleviate this task, instead of performing a full

all-pairs analysis, we drastically narrow down the analysis space and limit the correspondence estimation to similar pairs of shapes, which is a more tractable problem. To obtain a full correspondence estimation across the entire collection, we determine possible inference routes by estimating shape similarity and correspondence simultaneously. Correspondence inference then advances through a front of similar shapes and incrementally spreads throughout the collection. Assuming the collection is sufficiently dense, local similarity along the routes is expected to be high, thereby increasing robustness by reducing the risk of an ambiguous and potentially misleading correspondence.

The correspondences are established via a structure-oriented matching method, designed to partition a source shape into regions with similar structure, and match them to corresponding regions in a target shape, providing a part-based estimation of similarity between entities. As we focus on uncovering similar shapes, our matching does not provide a general solution that is meant for any possible shape pairing. Instead, the process supports and prefers a partial matching of high compatibility in favor of a potentially questionable full matching. A structural difference is therefore an event inherently recognized by the method, and is an immediate warning sign for dissimilarity. Our key concept is thus to leverage these differences to ensure a more reliable advancement in correspondence inference, by adding a fine-grained notion of structural similarity to the connections of the network. An interesting by-product of this approach is the partitioning of the collection into groups of shapes with similar structure. The shape collection can hence be represented by a structural similarity graph, where a node represents a group of shapes with similar structure and an edge represents a change in structure (Figure 1).

A central contribution of our approach is its robust pairwise correspondence computation. We bypass the difficulty involved in estimating a correspondence between two shapes in the 3D domain by solving a large set of simple 1D correspondence problems instead. Inspired by the projective shape analysis method [Wang et al. 2013], we first obtain numerous projections of a shape in 2D. Each projection then undergoes a partitioning which produces a 1D succession of micro-parts of the projection. The matching between two projections is conducted in 1D, allowing for a coarse correspondence to be inferred in 2D. The result is that each 2D projection is implicitly partitioned into regions with similar structure, which often reflects on the actual 3D part partitioning of the shape. In this manner, we are able to identify the structural differences between shapes with greater ease, and simultaneously avoid the difficulty involved in a direct partitioning of a shape into parts. Moreover, the matching process is robust in its tolerance of affine transformations between matched parts. These projection-specific correspondences are then aggregated and lifted back into 3D to obtain a sparse face-level correspondence between the source and target shapes.

We demonstrate that, with this method, we are able to establish correspondences among the shapes in large collections with high diversity, and build a network of correspondence where information can be reliably propagated. We present an analysis of networks obtained from large collections of shapes, as well as different applications that benefit from such a network, such as segmentation transfer and shape retrieval.

2 Related Work

Shape similarity and correspondence, and the combination of the two in a scheme that spreads linearly through a large collection of shapes, are at the core of our method. In this section, we discuss relevant work in these areas.

Shape correspondence. The problem of determining a correspondence between two shapes can take on many forms. Different applications call for different types of correspondence, as surveyed by van Kaick et al. [2011b]. Here, we are interested in semantic correspondence, where connections between regions with a similar purpose are sought. Several methods have focused on exactly this problem. Among the recent efforts are consistent segmentation methods aimed at establishing part-wise correspondence between shapes across an entire collection. Supervised methods call for a set of labeled examples to be given as input, which determine what new examples can be segmented by geometric similarity [Kalogerakis et al. 2010; van Kaick et al. 2011a]. Semi-supervised methods use correspondence cues given by a user to infer a set-wide consistent segmentation [Wang et al. 2012]. Unsupervised methods utilize different tools, such as diffusion maps [Sidi et al. 2011], joint optimization [Huang et al. 2011], and subspace clustering [Hu et al. 2012], to compensate for the lack of knowledge otherwise supplied by labeled examples. The above methods either require a large amount of labeled data, or are not scalable to very large shape collections with variation in geometry and structure.

Laga et al. [2013] propose an unsupervised method to establish a part-based correspondence between two shapes, where the correspondence is determined with part-to-part distances containing both geometric part similarities and the similarity of part context within the shape. The focus given to the context of a part facilitates matching parts from two shapes under differences in geometry and structure. As such, it is essentially invariant to structural changes that can be disregarded when dealing with a small set of shapes. However, these differences are at times crucial to the correct inference of correspondences within a large collection. Recently, Alhashim et al. [2015] present an optimization scheme that searches for a minimal energy deformation between two shapes, from which a pairwise correspondence is inferred. This method emphasizes structural preservation while allowing topological differences and therefore performs well even under significant topology-based deformations. However, it is designed primarily for pairs of shapes rather than large collections, and relies on a high-quality per-shape segmentation.

Most relevant in our context are approaches that infer point-to-point correspondence maps among all the shapes in a set. Nguyen et al. [2011], Huang et al. [2012] and Huang and Guibas [2013] refine maps between shapes by enforcing cycle-consistency among multiple correspondences. The consistency is optimized with techniques such as diffusion-and-sharpening on a set of base shapes, and semidefinite programming. Kim et al. [2013] extract a part-based correspondence for a set of shapes and use that to derive point-to-point correspondences. The part-based correspondence is obtained by fitting a set of box templates to a collection of shapes, which possibly involves user input for template creation. Finally, Huang et al. [2014] revisit cycle-consistency by optimizing it through functional maps, which serve as an efficient correspondence representation and allow to encode maps at multiple levels of detail. We observe that these methods adapt to changes in the structure of shapes in the collection, e.g., by spanning the creation of new templates [Kim et al. 2013], or with partial matching [Huang et al. 2014], but do not aim at detecting structural changes. In contrast, our method recognizes structural changes and uses them to build a network of reliable correspondences. The implicit grouping that is formed in the network captures informative fine-grained structural variations within a collection.

Shape similarity and comparison. Determining the extent of similarity between two shapes is crucial in applications such as shape retrieval and ordered exploration [Tangelder and Veltkamp 2008]. The notion of similarity is highly dependent on the con-

text or requirements of the application. Isometrically-related shapes can be compared and matched via appropriate signatures, such as the Heat Kernel Signature (HKS) [Ovsjanikov et al. 2010], GPS embedding [Rustamov 2007], blended intrinsic maps [Kim et al. 2011], or bag-of-feature descriptors [Bronstein et al. 2011; Litman et al. 2014]. Recently, Solomon et al. [2016] use a regularized Gromov-Wasserstein mapping objective to extend traditional correspondence methods to minimize long-range distance distortions, in addition to the typical local considerations used by these methods. Such an approach enables successful correspondence of a larger variety of shapes, and naturally supports multi-modal correspondences. As we will demonstrate, these methods are robust under non-rigid transformations (e.g., humanoid poses), but are not suitable for comparison of man-made shapes with structural variations.

Comparison methods that are based on the Light Field Descriptor [Chen et al. 2003] are suitable for rigid shapes with little to no articulation, and have been shown to perform well under these conditions [Su et al. 2015; Li et al. 2015]. Kleiman et al. [2015] suggest a part-based shape comparison approach, which defines the dissimilarity between shapes as the amount of effort it takes to transform one shape into the other. This method is robust in its invariance to shape articulation and is therefore suitable for both man-made as well as natural shapes. However, as it relies on a sufficiently good estimation of 3D segmentation, it is not immediately scalable to large shape collections.

Networks of similarities. Correctly determining the similarity of data points in a high-dimensional space has been the focus on much research in statistics and machine learning. A common idea to address this problem is to compute local similarities between points, which are then used to robustly estimate the similarities of points that are far in the space, revealing the intrinsic manifold on which the points reside. For example, isomaps derive the similarity of points from their distance within a graph of nearest neighbors [Tenenbaum et al. 2000], while diffusion maps perform a spectral embedding that captures the commute distance between points [Coifman and Lafon 2006]. We follow the same general principle in our work, where we connect highly similar shapes to create a network that spans the entire collection. However, in addition to the network, each shape connection possesses correspondence information that maps the geometry of one shape to another.

3 Similarity-driven correspondence

Our aim is to establish correspondences within large-scale shape collections, where shapes often exhibit considerable variations in geometry and structure. In this context, it is impractical to employ an all-pairs correspondence computation approach. Thus, we limit the computation of correspondences to similar shapes. This approach is advantageous not only for its reduced complexity, but also, and even more importantly, because of the increase in reliability. A correspondence estimated between a pair of sufficiently dissimilar shapes is inherently subject to errors, but incorrect matches are less likely to occur between two similar shapes. We therefore develop a robust correspondence estimation method that is geared toward similar shapes. With such a method in hand, we are able to form a network of correspondences spanning the entire collection, and identify reliable correspondence inference routes supporting propagation of information between shapes.

To compute a reliable correspondence between two shapes, we opt to emphasize structural differences as major cues for dissimilarity, since these differences are often a primary obstacle in correspondence estimation. This assumption requires a method to detect structural differences between two shapes, possibly through a structural decomposition. A common approach for obtaining such

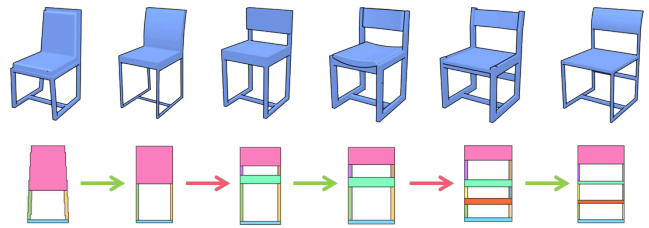


Figure 2: Structural changes along a path of correspondences. The shapes (top row) are projected from the back viewpoint into low-resolution images, shown beneath. Each projection is first decomposed into regions that are then matched to the regions of the next projection in the sequence. The matched regions are colored consistently and illustrate a regional correspondence that is propagated along a sequence of shapes. Note that while the matching is conducted on 2D projections, the correspondence information resulting from this process is aggregated and lifted back to the 3D domain, where it is propagated from and to entities in the collection. We show here a 2D illustration of a possible path of correspondence propagation, highlighting two structural changes that are discovered by the matching process (red arrows). The first event is caused due to a change in the topology of the back of the chair, and the second marks the addition of a bar connecting the two legs.

a decomposition is to perform a skeletonization of the geometry of a shape. However, in the presence of complex geometry, the accuracy, reliability and efficiency of such methods are low. We introduce an alternative approach and reduce the problem to the 2D domain by analyzing multiple projections of the shape.

Our 2D analysis is inspired by the work of Wang et al. [2013] on projective shape analysis, but applies this paradigm to the problem of shape partitioning and matching. We carefully partition each projection of a shape according to shape structure, based on a set of rules that consider various structural cues. This process partitions the 2D projection into a 1D ordered set of regions, such that the notoriously hard problem of graph matching [Kleiman et al. 2015] can be avoided. The simplicity of this approach also supports meaningful partial matching, which is crucial for the detection of structural differences between two shapes. Figure 2 presents an overview of our correspondence inference approach, depicting a sequence of matching steps uncovering important structural changes between shapes along a possible correspondence propagation path. We describe the matching approach in more detail in Section 4.

Prior to forming a similarity-based network of correspondences, we compute a rough estimate of shape similarity with signatures derived from HoG-based Light Field descriptors. Next, each shape is paired with its k most similar shapes according to these signatures, and correspondence is established for each pair via our structure-oriented matching. By choosing an appropriate value for the parameter k , we balance between computation cost and network quality. A small value for k signifies a reduced set of shape pairs undergoing the matching process, at the cost of a simultaneous reduction in the amount of potentially useful direct connections that the network of correspondences relies on. Finally, we leverage similarity scores computed throughout the matching process to identify quality pairwise correspondences that define reliable connections within our correspondence network. These connections can then be used to compute a route of information propagation between shapes in the collection, as described in Section 5.

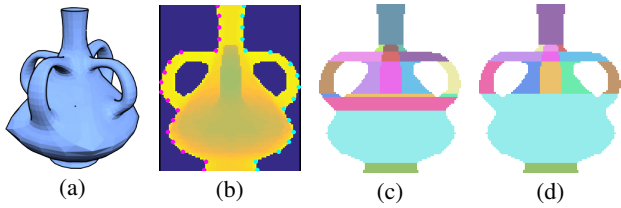


Figure 3: Region decomposition, where the depth projection (b) of a vase (a) is analyzed to obtain the 2D decomposition into regions shown in (d). An initial division into regions (c) is achieved by merging together adjacent rows based on depth and structural similarity, as well as shape convexity, estimated by the angles between consecutive triplets of sampled points (in pink and cyan in (b)). These initial regions are further merged into more complex regions via a relaxation of the merging conditions, to form the final decomposition (d), consisting of meaningful regions that can be leveraged for shape matching.

4 Structure-oriented shape matching

Our input is a collection \mathcal{S} of pre-aligned shapes. We project each input shape $S_i \in \mathcal{S}$ from n viewpoints (4-6 in our experiments) to obtain a set of depth projections $P_i = \{p_k^i\}_{k=1}^n$. We use the z -buffer algorithm for this step, where each projection is an image of size $h \times w$, and pixel values are in the range $[0, 1]$. We denote a pixel with value 0 as an empty pixel, i.e., no portion of the shape was projected onto it, and the value $\varepsilon > 0$ is assigned to the pixels that are closest to the shape, such that no pixels have a value in $(0, \varepsilon)$. Since the shapes are pre-aligned, we assume that projection perspectives match across shapes, i.e., p_k^i for S_i corresponds to p_k^j for S_j . If the shapes are not aligned, we can pre-process the set to align the shapes based on a matching of the projections.

Given two similar shapes, the input to the structure-oriented matching step is a set of 2D projections for each shape. To compute a correspondence between the shapes, we establish a correspondence between each pair of their 2D projections by solving a 1D matching problem. We first decompose each image into a sequence of regions based on differences in shape structure and projection depth, which possibly define boundaries between semantic shape parts (Section 4.1). Next, the decompositions from the two shapes are matched with a dynamic programming algorithm, yielding a correspondence between regions and an estimation of the similarity between the two shape projections (Section 4.2). Finally, we aggregate the information provided by each projected view by lifting the matches from the 2D domain to obtain a 3D correspondence (Section 4.3).

4.1 Region decomposition

The goal of the decomposition is to partition each projection into a set of regions that consist of simple structures and form a 1D vertical sequence. The advantage of using such a partitioning is two-fold. First, region boundaries that follow the shape structure form a structure-oriented decomposition, providing a strong indication of the semantic partitioning of the shape. Second, a one-dimensional decomposition greatly simplifies the task of matching two projections, as the problem is reduced to matching two linear sequences. Since we emphasize structural differences as an important indication of shape similarity, we require a meticulous analysis of the shape structure reflected by the projections.

In our work, each region of the 1D vertical sequence is potentially composed of multiple segments laid out horizontally. The *regions*

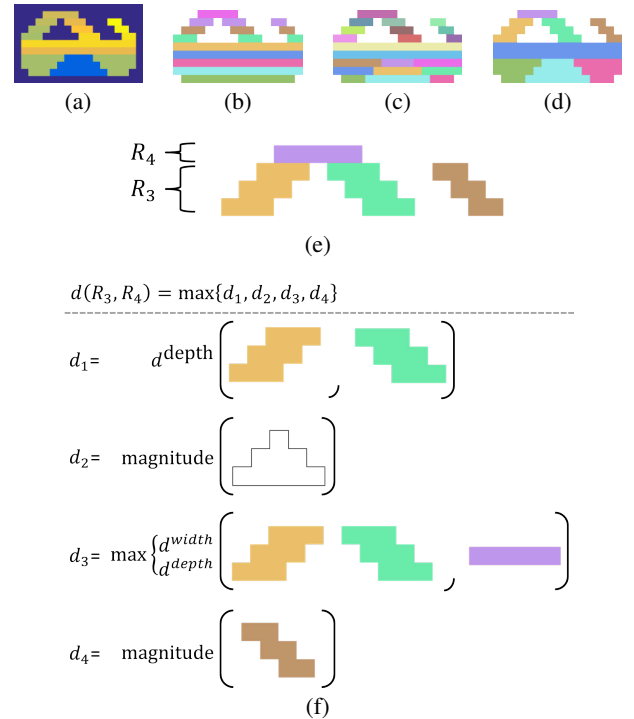


Figure 4: A simple region decomposition example. A depth projection (a) is analyzed on a per-row basis (b), to form initial row regions containing segments with boundaries determined by holes or significant depth changes, shown in (c). In (d), four primitive regions are formed by merging adjacent regions from (c), following rules designed to preserve inner region structure and depth. In (f), we examine the cost of merging R_3 and R_4 (shown close-up in (e)). This union incurs penalties for joining separated segments (d_1 and d_2) and segment discontinuity (d_4), aside from the basic size and depth dissimilarities (d_3).

are created by concatenating together rows of pixels with similar structure and depth, and each region is internally broken into *segments* based on discontinuity in the form of holes (one or more sequential empty pixels), or significant depth changes; see Figure 3 for an illustration of the method.

Our decomposition is obtained in two steps. First, we create a set of primitive regions that serve as good building blocks. Next, these regions are merged into more complex ones, while preserving the required properties of the decomposition: maintaining a 1D sequence of regions with consistent depth and structure. This two-step approach allows better control over the decomposition and the nature of the obtained regions. It provides a balance between the structural complexity within the regions and the overall complexity of the decomposition, which is difficult to achieve with a greedy approach.

An alternative method to decompose a projection of a 3D shape is based on the Hausdorff distance [Wang et al. 2013]. This method, however, forms regions that do not necessarily capture shape structure, as it relies on the maximum distance between empty and non-empty pixels, without accounting for contiguity or depth changes. See Figure 8 for a visual comparison, and Section 7 for a quantitative one.

Primitive regions. In the first step of the decomposition, a projection p_k^i is decomposed into regions by consistently merging adjacent rows of the depth image. Each row is initialized as a re-

gion and is divided into a set of segments according to structural changes within the row. Segment boundaries are induced by significant change in depth (> 0.2 in our experiments) or by holes, as depicted in Figure 4(a-c).

Next, neighboring regions are merged together on a pairwise basis to create larger regions if four conditions apply: (i) The regions have the same number of segments; (ii) The segment columns overlap, indicating that the segments are contiguous; (iii) The two regions are contained within the same weakly-convex part; and (iv) The region segments have similar depth and size. Since merging is performed on a pairwise basis, we can efficiently traverse the rows of the image in their natural order, and introduce a boundary between any two rows that do not comply with the conditions.

We note that conditions (i) and (ii) guarantee a one-to-one pairing between the segments of two adjacent rows. Thus, to enforce condition (iv), we simply compare the lengths and depths of the matched segments. Segment length is computed as the number of pixels in the segment divided by projection width, and segment depth is the median among the segment depth values. The dissimilarity between two segments is then defined as the maximum between length and depth discrepancies, and the merge cost between two adjacent rows is taken as the maximum between all segment dissimilarities. Condition (iv) is satisfied when this merge cost does not surpass a threshold $\theta_{\text{sim}} = 0.12$ (determined empirically).

Finally, condition (iii) is enforced in order to identify concavities in the silhouette of the shape that do not manifest as changes in structure or depth. Rather than performing an in-depth convexity analysis of the shape, we efficiently identify part-aware cues for our decomposition by computing a vertical division of the shape into weakly convex parts. We sample points on the shape silhouette from top to bottom on the left and right boundaries (see Figure 3(b)), and compute the angle that is formed by any consecutive triplet. Aiming to disregard small concavities, we determine that any angle that is smaller than 165° is a sufficiently concave angle, and set a boundary between the image rows below and above it.

At the end of this process, we obtain a set of primitive regions composed of segments that are approximately convex and with consistent depth (see Figure 3(c) and Figure 4(d)). Next, we create more complex regions by further merging adjacent regions through a relaxation of the merging conditions.

Final regions. In this step of the decomposition, we suspend the first three merging conditions and greedily merge regions as long as their similarity in terms of depth and structure is below a threshold $\theta_{\text{Merge}} = 0.12$. However, merging structurally-different regions requires special handling, since there is no longer a one-to-one matching between the segments of the two regions. To this end, we merge any two segments that are contiguous. This approach leads to the merging of separated segments in a region, if there exists a segment in the neighboring region that connects them. For example, in Figure 3(c), the separate segments of the fourth region from the bottom are all contiguous with the single segment of the region below them. Therefore, a merge between these two regions requires not only a merge between segments across regions, but also between segments within the same region, as illustrated in (d).

To maintain a certain level of structural coherence within the regions, we extend our region merge cost beyond depth and segment size, and incur a cost for any two separated segments that were merged, as well as for any segment in one region that is not continued in the merged region. We define the cost of joining two separated segments as the maximum between the amount of empty space between them, and their depth discrepancy. We note that at this stage a region may be composed of multiple rows, and each segment may therefore be composed of several row segments of vary-

ing sizes. We define the size of a segment to be the average width of its row segments. Accordingly, the size of the empty space area separating two segments is set as the maximum between the height and the average width of the empty space.

We penalize a merge that does not follow shape structure according to the severity of the structural violation, i.e., by attaching a larger cost to a merge between two well-separated segments. This is made possible by our two-step approach, where primitive regions with similar structure are first formed, allowing an informed examination of the extent of structural violations. In Figure 3(c), we note that a merge between the third and fourth regions (from bottom) leads to a minor structural violation, unlike a merge between the fifth and sixth regions – a severe violation that is therefore avoided in (d).

To account for a segment in one region that is not continued in the region it is merged to, we take its magnitude (the maximum between its height and average width) as the cost of its discontinuity, once again emphasizing the importance of substantially sized segments. We define the merge cost between two regions R_j and R_{j+1} as the maximum among all costs pertaining to their union.

In Figure 4(f), we detail the set of costs associated with the potential merge of regions R_3 and R_4 . The purple segment in R_4 is contiguous to both the orange and green segments in R_3 , implying that these three segments will become one in $R_{3 \cup 4}$. The joining of the orange and green segments results in two penalties – their depth discrepancy (d_1) and the extent of separation between them (d_2). d_3 is simply the cost of merging segments from two regions, and d_4 penalizes for the discontinuation of the brown segment in R_3 .

By greedily merging regions as long as the cost is below θ_{Merge} , we obtain the final decomposition, where the regions \mathcal{R}_k^i of projection p_k^i are ordered vertically, and each region contains a set of segments that are ordered horizontally (see Figure 3(d)). We denote the decomposition of projection p_k^i by $\mathcal{R}_k^i = \{R_m\}_{m=1}^{M_k^i}$ (M_k^i is the number of regions in p_k^i), where $R_m = \{\text{seg}_l\}_{l=1}^{L_m}$ (L_m is the number of segments in R_m). Here, $\text{seg}_l = [h^{\text{PCA}}, w^{\text{PCA}}, \text{axis}^{\text{PCA}}, \text{dep}]$. That is, each segment is represented by its height, width and main axis as given by a PCA analysis of its shape, as well as its median depth. This representation will be utilized for matching in the next section.

4.2 Region matching

Given the region decompositions \mathcal{R}_k^i and \mathcal{R}_k^j for the projections p_k^i and p_k^j , respectively, obtained from view k of shapes S_i and S_j , we seek to find an optimal matching of the regions. We allow the matching to be partial, preferring a region or segment to be left unmatched rather than to force a bad match. This policy increases our trust in the inferred matches and supports our goal of detecting highly similar shapes. The matching is computed with a dynamic programming scheme based on the cost of matching two regions.

Segment matching. Since the regions are composed of segments, a matching of two regions amounts to a matching between their constituent segments. We match segments in an order-preserving manner, and allow the matching to be partial. Given $R_m \in \mathcal{R}_k^i$ and $R_n \in \mathcal{R}_k^j$, such that $R_m = \{\text{seg}_x\}_{x=1}^M$ and $R_n = \{\text{seg}_y\}_{y=1}^N$, the sets of segments are matched using a dynamic programming approach governed by the recursive equation:

$$M_{\text{segs}}(s_i, s_j, x, y) = \begin{cases} d_{\text{NM}}(s_j[y : \text{end}]), & \text{if } x > \|s_i\|, \\ d_{\text{NM}}(s_i[x : \text{end}]), & \text{if } y > \|s_j\|, \\ \min\{c_1, c_2, c_3\}, & \text{otherwise,} \end{cases} \quad (1)$$

$$\begin{aligned}
\text{with } c_1 &= d_{\text{seg}}(s_i[x], s_j[y]) + M_{\text{segs}}(s_i, s_j, x + 1, y + 1), \\
c_2 &= d_{\text{NM}}(s_i[x]) + M_{\text{segs}}(s_i, s_j, x + 1, y), \text{ and} \\
c_3 &= d_{\text{NM}}(s_j[y]) + M_{\text{segs}}(s_i, s_j, x, y + 1).
\end{aligned} \tag{2}$$

Throughout the matching process, we compare $\text{seg}_x \in R_n$ against $\text{seg}_y \in R_m$ using the segment properties. More specifically, we set the dissimilarity between the two segments, denoted by d_{seg} , as the maximum between the width, height, orientation and depth differences of the segments. Similarly to the penalty given to discontinued segments during decomposition, a segment that is left unmatched is penalized by $d_{\text{NM}}(\text{seg}) = \max(h^{\text{PCA}}, w^{\text{PCA}})$.

Region matching. Given the segment matching cost, we can now define the cost of matching two regions. We use this cost to obtain a correspondence between two sets of regions. The 1D ordering of the regions allows us to perform a simple in-order region matching, much like the approach adopted for segment matching (Equation 1).

To support partial matching at the region level, we empirically determine a threshold $\theta_{\text{Sp}} = 0.3$ to signify a subpar region pairing. This threshold value is then used as the penalty given to unmatched regions, to ensure preference of partiality over low compatibility. Thus, we define the region matching process as follows:

$$M_{\text{regs}}(\mathcal{R}_k^i, \mathcal{R}_k^j, x, y) = \begin{cases} \theta_{\text{Sp}} * (|\mathcal{R}_k^j| - y + 1), & \text{if } x > |\mathcal{R}_k^i|, \\ \theta_{\text{Sp}} * (|\mathcal{R}_k^i| - x + 1), & \text{if } y > |\mathcal{R}_k^j|, \\ \min\{c_2, c_3\}, & \text{if } c > \theta_{\text{Sp}}, x \leq |\mathcal{R}_k^i|, y \leq |\mathcal{R}_k^j|, \\ \min\{c_1, c_2, c_3\}, & \text{otherwise,} \end{cases} \tag{3}$$

$$\begin{aligned}
\text{with } c &= M_{\text{segs}}(\mathcal{R}_k^i[x], \mathcal{R}_k^j[y], 1, 1), \\
c_1 &= c + M_{\text{regs}}(\mathcal{R}_k^i, \mathcal{R}_k^j, x + 1, y + 1), \\
c_2 &= \theta_{\text{Sp}} + M_{\text{regs}}(\mathcal{R}_k^i, \mathcal{R}_k^j, x + 1, y), \text{ and} \\
c_3 &= \theta_{\text{Sp}} + M_{\text{regs}}(\mathcal{R}_k^i, \mathcal{R}_k^j, x, y + 1).
\end{aligned} \tag{4}$$

Output. A completed matching process provides us with a set of matched segments $\mathcal{M}_{ij} = \{\langle \text{seg}_t^i, \text{seg}_t^j, \sigma_t \rangle\}_{t=1}^{T_m}$, where σ_t is the matching score of the pair of matched segments $\langle \text{seg}_t^i, \text{seg}_t^j \rangle$ (see Figure 5). In addition, as a result of our partial matching, we obtain two sets of segments $\mathcal{N}_i = \{\text{seg}_t^i\}_{t=1}^{T_{n_i}}$ and $\mathcal{N}_j = \{\text{seg}_t^j\}_{t=1}^{T_{n_j}}$, collecting the segments from each projection that are left unmatched (see Figure 5(m1), (m1') and (m2)). Each of these sets may be empty following a two-sided or a one-sided full match. These unmatched segment sets play a crucial role in the detection of structural differences between shapes, as will be described in the following sections. To estimate and summarize the similarity between the pair of projections of view k , we define the matching score of the projections, $\mu_{i,j}^k$, as the mean of all σ_t , and the overall no-match scores, $\nu_{i,j}^k, \nu_{j,i}^k$, as the ratio of unmatched area to the overall projection area of p_k^i and p_k^j , respectively.

Note that our scheme differs from the one presented by Wang et al. [2013] by a number of factors. Here, the regions are created with the objective of preserving structure. Thus, the number of regions is not predetermined, but is an outcome of the structure-preservation objective. As such, our decomposition contains well-formed segments that can be compared based on their structural properties. Additionally, we monitor and prevent any subpar region matches that may jeopardize the reliability of the matching and subsequent correspondence inference.

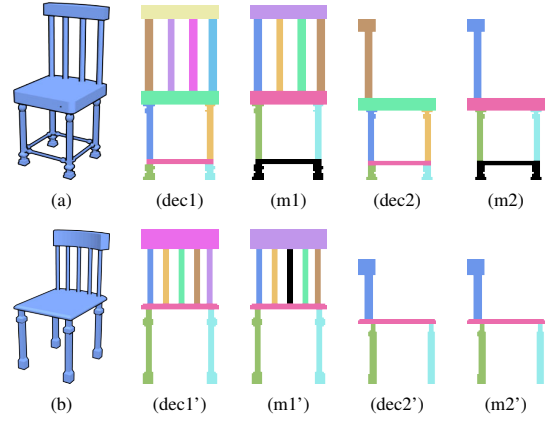


Figure 5: Region matching of shape projections. To obtain a correspondence between shapes (a) and (b), we take their region decompositions for two projections ((dec1) + (dec1') and (dec2) + (dec2')). We apply our matching process on the decompositions and obtain the correspondences between (m1) and (m1'), and between (m2) and (m2'). Matched segments are colored consistently, while unmatched segments are shown in black.

Please refer to our supplementary material for visual examples of shape decomposition and matching.

4.3 View aggregation for 3D correspondence

In order to infer a correspondence between S_i and S_j in the 3D domain, we aggregate the matches obtained for each view. We translate each pair of matched projections into a correspondence between faces of the 3D model as follows.

A pair of matched projections contains a set of matched segments, where each segment is essentially a collection of pixels within a projection. Thus, we derive a pixel-level correspondence between a pair of projections through their matched segments. For matching segments, each non-empty pixel in one segment is matched to its nearest non-empty pixel in the other, taking relative differences in size into account. We construct a matrix \mathcal{A}_k of size $x_k^i \times x_k^j$, such that x_k^i and x_k^j are the total number of pixels in p_k^i and p_k^j , and set $\mathcal{A}_k(\text{px}_m, \text{px}_n) = 1$ for any pair of matched pixels.

Next, let \mathcal{F}_k^i and \mathcal{F}_k^j be two matrices of size $f^i \times x_k^i$ and $f^j \times x_k^j$ respectively, such that f^i and f^j are the number of faces in shapes S_i and S_j . These matrices are formed during the projection process and record the correspondence between shape faces and projection pixels, such that $\mathcal{F}_k^i(\text{fc}_m, \text{px}_n) = 1$ when face $\text{fc}_m \in S_i$ is projected onto pixel $\text{px}_n \in p_k^i$, and is the closest one to the projection plane out of the set of faces that are projected onto px_n . Note that a certain pixel may be associated with multiple faces, all of which share the closest distance to the plane.

We utilize the three matrices $\mathcal{F}_k^i, \mathcal{F}_k^j$ and \mathcal{A}_k to obtain a correspondence between the faces of S_i and S_j for view k by a simple matrix multiplication: $\mathcal{C}_k^{ij} = \mathcal{F}_k^i \mathcal{A}_k \mathcal{F}_k^{jT}$. The aggregated correspondence from all projection views is then given by the summation: $\mathcal{C}^{ij} = \sum_{k=1}^n \mathcal{C}_k^{ij}$. Due to the many-to-many relation that exists between faces and pixels, the matrix \mathcal{C}^{ij} is not simply a 0/1 matrix, but rather contains values indicating, for each pair of matched faces, the number of times they were matched through a pair of pixels (in matrix \mathcal{A}_k). In Section 7 we utilize this information as confidence weights for segmentation transfer.

In addition to obtaining a 3D correspondence between S_i and S_j , we summarize the matching and no-match confidences collected from all views, as they provide a measure of similarity and reliability of the inferred correspondence. We define the matching score between S_i and S_j , denoted by $\mu_{i,j}$, as the mean of all $\mu_{i,j}^k$, and the no-match ratios, denoted by $\nu_{i,j}$ and $\nu_{j,i}$, as the mean of all $\nu_{i,j}^k$ and $\nu_{j,i}^k$, respectively. In the following section, these measures will play a central role in the creation of a network of correspondences spanning the entire collection of shapes.

5 Correspondence inference within a large collection

To establish correspondences among the shapes of a collection, we identify pairs of similar shapes and establish correspondences only between them. The correspondence between two dissimilar shapes can then be derived from a chain of correspondences between shapes that were ruled as similar.

The first task in support of correspondence propagation within a collection is thus to search for pairs of similar shapes. Our structure-oriented matching approach, described in Section 3, helps us to identify problematic shape pairings. By forbidding any sub-region matches, we detect correspondences that are prone to ambiguity, which is reflected in their matching scores. We are then able to eliminate or ignore these questionable shape pairings when constructing and using our network of correspondence. However, despite the efficiency of our matching process, it is computationally expensive to perform an analysis of this kind on all pairs of shapes in collections of large magnitudes. Thus, we narrow down our comparison space considerably prior to establishing any correspondences within the collection.

We extract a set of candidate pairs of similar shapes efficiently with the similarity computation introduced by Li et al. [2015]. This approach is based on the Light Field Descriptor [Chen et al. 2003] of a shape. Two shapes S_i and S_j are projected from k viewpoints, and their dissimilarity is estimated as $d(S_i, S_j) = \|HoG_i - HoG_j\|$, where HoG_i and HoG_j are the concatenation of the Histogram of Gradient descriptors [Dalal and Triggs 2005] for all k viewpoints. Let us denote by $\mathcal{D}_{n \times n}$, the all-pairs distance matrix, such that $\mathcal{D}(i, j) = d(S_i, S_j)$. From \mathcal{D} , we select, for each shape S_i , its K -nearest neighbors, denoted by KNN_i .

We compute our structure-oriented correspondence between S_i and each $S_j \in KNN_i$, and obtain a tuple $\langle \{\mathcal{M}_{ij}, \mu_{i,j}, \nu_{ij}, \nu_{ji}\} \rangle$, containing the matched segments, matching score of the shapes, and no-match ratios for each shape in the pair. We create three matrices to capture the similarity and structural compatibility within the collection: \mathcal{D} , \mathcal{C}_d and \mathcal{C}_s . We set $\mathcal{D}(i, j) = \mathcal{D}(j, i) = \mu_{i,j}$, $\mathcal{C}_d(i, j) = \nu_{ij}$, $\mathcal{C}_d(j, i) = \nu_{ji}$, and $\mathcal{C}_s(i, j) = \mathcal{C}_s(j, i) = \nu_{ij} + \nu_{ji}$. That is, \mathcal{D} summarizes the pairwise distances between shapes, and \mathcal{C}_d summarizes the directed incompatibility between a pair of shapes, while \mathcal{C}_s is its symmetric version where the sum of incompatibilities is taken. These matrices essentially define a connectivity graph with weighted edges spanning the entire collection, such that only the edges defined by \mathcal{C}_d are directed.

To obtain a correspondence between an arbitrary pair of shapes S_i, S_j , we search for a path of inference through third parties in the graph and compute the correspondence matrix \mathcal{C}^{ij} by multiplying the correspondence matrices of shapes along the path. One possibility to derive an inference path is to search for the minimum cost path from S_i to S_j within the connectivity graph, for instance, by considering the sum of weights given by \mathcal{D} and \mathcal{C}_s . However, such a path may still include edges with large weights, as only the sum of weights is minimized. Thus, we favor a path for which the weight of

each edge along the path is minimal, contributing to its overall reliability. A path with this property is known as a *minimax path*. In an undirected graph, such as the one given by $\mathcal{G}_s = \mathcal{D} + \mathcal{C}_s$, the minimax path is the path from S_i to S_j in the minimum spanning tree of \mathcal{G}_s , denoted by $\mathcal{T}^{\mathcal{G}_s}$. In the directed graph given by $\mathcal{G}_d = \mathcal{D} + \mathcal{C}_d$, this path can be found through a modification of Dijkstra’s shortest path algorithm [Kaibel and Peinhardt 2006]. For general correspondence inference between two shapes, there is no significance to direction, since correspondence is bidirectional. However, in the following sections, we will evaluate the practical impact of edge directionality in applications such as segmentation propagation.

6 Experiments and evaluation

In this section, we analyze the result of applying our method on shape collections of varying sizes for the construction of correspondence networks. In Section 7, we demonstrate how these networks can be used in diverse applications, and perform additional evaluation related to these applications.

Datasets. Recently, large collections of shapes have started to form, following efforts to gather models from various available sources and group them into semantic categories [Chang et al. 2015]. We evaluate our method on nine sets of shapes. Of these, six are purely from the ShapeNet collection [Chang et al. 2015] (tables (8,509), cars (7,497), planes (4,045), lamps (2,318), guitars (797), faucets (744)), two are from the COSEG dataset [Wang et al. 2012] (chairs (400) and vases (300)), and the final set contains 7,178 chairs (6,778 from ShapeNet and 400 from COSEG). The sets obtained from ShapeNet are of considerable size and lack any correspondence or segmentation ground-truth information, therefore, we perform a qualitative analysis on them in order to evaluate the behavior of our method. Conversely, the COSEG sets are provided with ground-truth segmentations, allowing us to perform a quantitative analysis for this application; see Section 7. Ideally, we would also perform such an evaluation on the larger datasets that are the focus of our work, e.g., by collecting ground-truth information with crowdsourcing. However, we note that such ground-truth information is not always reliable, especially given the ambiguity in correspondence estimation and segmentation for general shapes. Perhaps ground-truth information should also capture a notion of shape similarity, where certain segmentations or correspondences are only applicable within portions of the set that are unambiguously related.

Correspondence network. To assess the quality of the networks of correspondences created with our method, we first inspect a few qualitative samples extracted from the networks. In Figure 6, we show six subtrees extracted from the minimum-spanning tree $\mathcal{T}^{\mathcal{G}_s}$ of entire datasets, where the trees are constructed as described in Section 5. We select a few shapes from different regions of the networks and use them as root nodes. Then, we extract the subportions of these trees spanning from the roots to a few levels in depth.

In these examples, we identify various manners in which the structure of the shapes is captured. When studying each individual tree, we note how the complexity of shape structure gradually changes as we traverse the edges away from the roots. For example, in (a), we detect changes in back structure, containing a varying number of bars, as well as changes in bars connecting the chair legs. Moving toward the leaves, we can even locate a few chairs with handles.

Please refer to our supplementary material for a quantitative comparison on the BHCP benchmark [Kim et al. 2013] and further examples featuring correspondence subtrees.

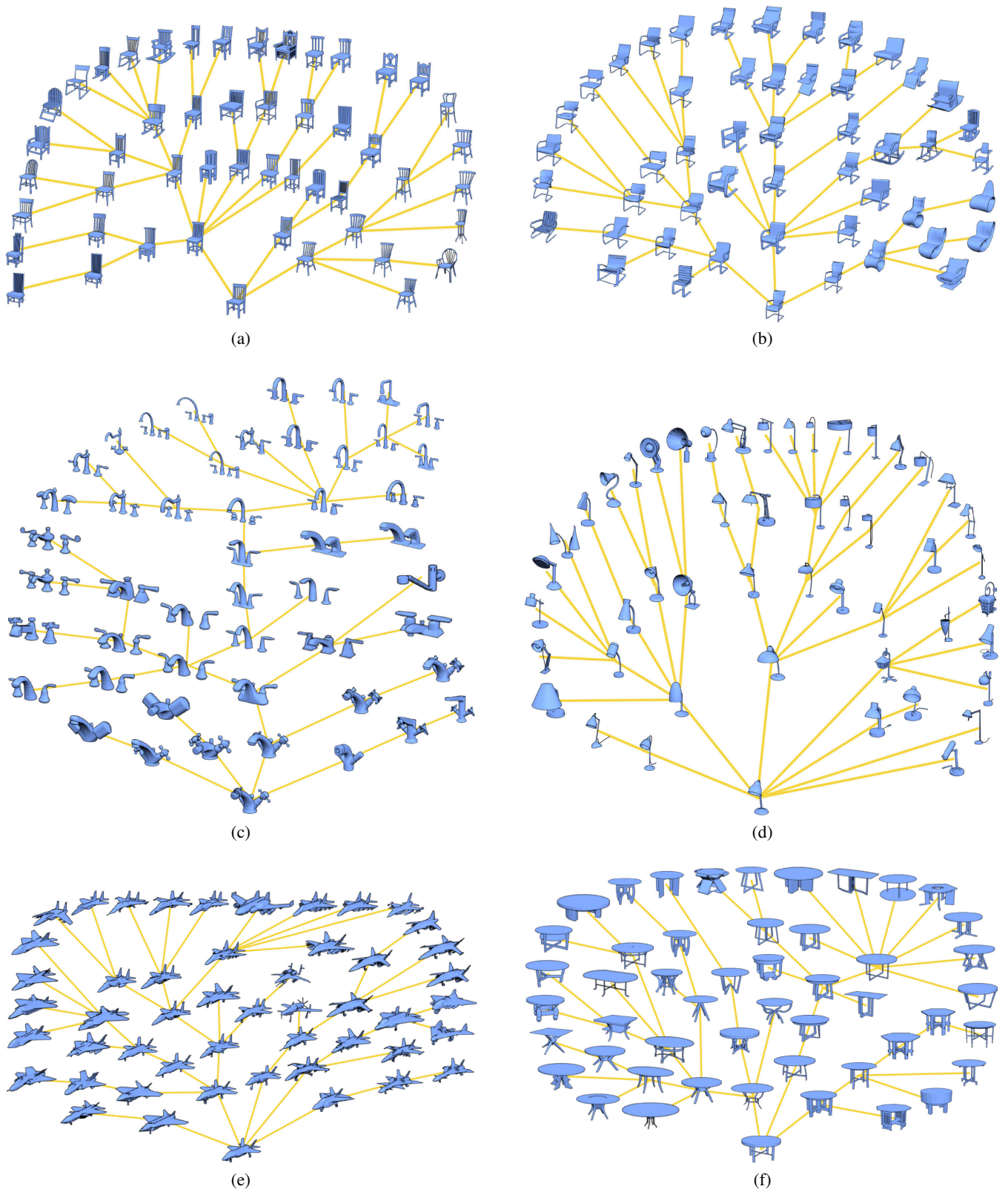


Figure 6: Selected subtrees sampled from our correspondence networks. Given a source shape (the root at the bottom of each tree), we show shapes that can be reached by following a few neighbor connections in the network of correspondences.

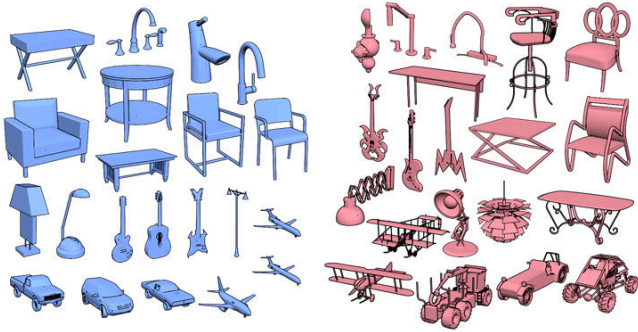


Figure 7: Shape influence. We show examples of highly influential shapes, in blue, vs. non-influential shapes, in red. An influential shape is a shape with a wide propagation reach, signifying that it can reliably transfer information to a large number of other shapes, either directly or by transitivity.

Shape influence. The directionality given to the structural differences between two shapes by the matrix C_d (Section 5) essentially captures relative differences in structural complexity. For a pair of shapes S_i and S_j , if S_i contains all the regions of S_j , but the converse is not necessarily true, then S_i is structurally more complex (or structurally equivalent) to S_j . If S_i is still sufficiently similar to S_j , despite a higher complexity, we determine that S_i has positive influence over S_j , signifying its potential ability to reliably transfer information to S_j . In this context, influence is also a transitive property. If S_i has influence over S_j , and S_j has influence over S_k , then by transitivity S_i has influence over S_k , and information can reliably propagate from S_i to S_k .

To determine the extent of influence, or propagation reach, of S_i , we perform an analysis using the data collected throughout the matching process. Aside from C_d , recall the undirected graph $\mathcal{G}_s = \mathcal{D} + \mathcal{C}_s$ and the MST $\mathcal{T}^{\mathcal{G}_s}$ defined in Section 5. We define a directed graph \mathcal{G}_d initialized with edges given by C_d and remove any edge $e_{i,j} \in \mathcal{G}_d$ for which the weight on $e_{i,j} \in \mathcal{G}_s$ surpasses a threshold $\theta_{\text{Diff}} = 0.15$, and such that $e_{i,j} \notin \mathcal{T}^{\mathcal{G}_s}$. The role of \mathcal{G}_d is to emphasize directional structural similarity through the weights on its edges. But, by removing any edge that signifies an unreliable affinity between two shapes, we maintain general similarity along the paths of \mathcal{G}_d . Keeping edges that are a part of $\mathcal{T}^{\mathcal{G}_s}$ ensures full directional reachability across \mathcal{G}_d .

We can now compute the minimax path from S_i to all other shapes in the collection through a modification of Dijkstra’s shortest path algorithm. By construction, a minimax path from S_i to S_j in \mathcal{G}_d achieves the minimal amount of structural additions, while sustaining general similarity along the route. We locate all the shapes in the collection for which the minimax distance from S_i is below a threshold $\theta_{\text{Str}} = 0.01$, indicating structural complexity in favor of S_i , and mark them as the subset of shapes that are reachable from S_i . These are essentially the shapes over which S_i has potentially positive influence, as they can be transitively reached with minimal ambiguity-causing additions to structure. By sorting the shapes in the collection according to the size of their set of reachable shapes, we determine the relative extent of influence of each shape, according to its position in the sorted list.

Figure 7 features a subset of the most influential shapes (taken from the top 10% shapes with the largest propagation reach), and a subset of the least influential shapes (in the bottom 10%). We observe how the influential shapes cover a variety of “standard” structures, although they are not necessarily trivial, implying that they influence other shapes that are variations of these structures. The least

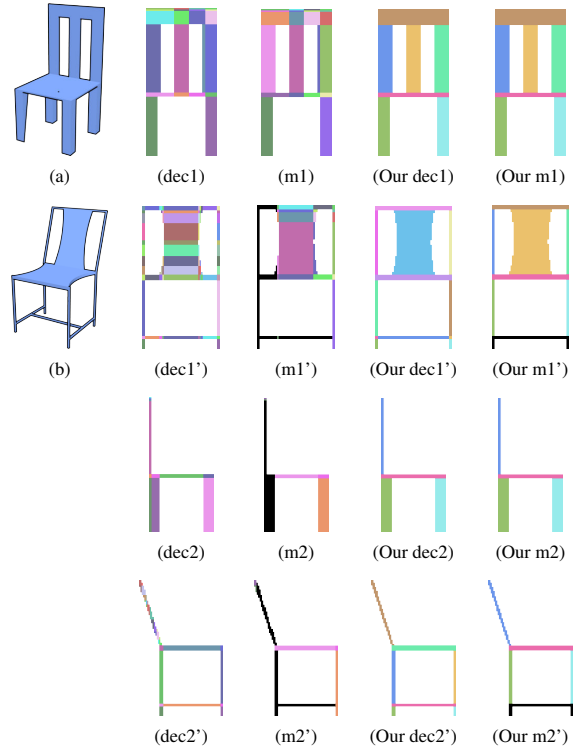


Figure 8: Comparison to Wang et al. We obtain two projections from the same viewpoint for the chairs in (a) and (b). The projections are then decomposed by the method of Wang et al. to form the decompositions (dec1) and (dec1’). The decompositions are then matched to each other, as shown in (m1) and (m1’) with corresponding region colors. In contrast, our method creates the decompositions (Our dec1) and (Our dec1’) and provides the matching shown in (Our m1) and (Our m1’), which is more structure-oriented and meaningful. Another comparison example is shown for (dec2).

influential shapes, however, have quite peculiar structures, which also appear less frequently in these datasets.

Comparison to Wang et al. In Section 4.2, we discuss the conceptual differences between our approach and that of Wang et al. [2013]. A visual comparison is shown in Figure 8, obtained by our own implementation of their method. In this example, two chair models are decomposed into horizontal and vertical slabs, and then matched. As instructed in their work, a target shape ((b) in the figure) is decomposed into twice the number of slabs than the decomposition of the source ((a) contains). We observe that, in comparison, our method yields decompositions that are more structure-oriented and meaningful, with less fragmentation.

7 Applications

Our network of correspondences spanning a collection of shapes can be leveraged for tasks that require reliable, similarity-driven correspondence inference within the collection. We present here a few example applications that benefit from such a network. Please refer to our supplementary material for more results.

Segmentation transfer. Large collections of shapes are likely geometrically and topologically varied. Thus, consistently segmenting them is challenging, as learning a set of rules to capture the properties of diverse shape parts is a complex problem. Manual

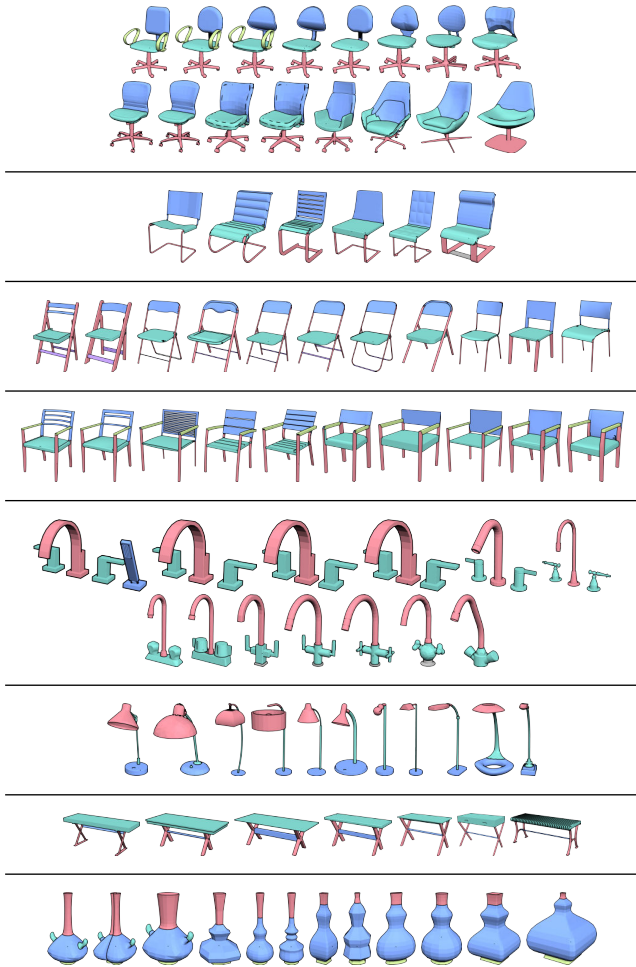


Figure 9: Segmentation propagation paths. Semantic segmentations are consistently transferred through face-level correspondences obtained by our matching process. Starting from a manual segmentation given for the leftmost source shape in a sequence, we propagate the segmentation along the path toward the rightmost target shape. Parts colored in grey have no match to the preceding shape and remain unlabeled.

segmentation is also impractical due to the significant workload involved. To alleviate this problem, we can make use of our network of correspondences for segmentation transfer, effectively segmenting the shapes in a semi-supervised fashion.

We start with a small set of segmented shapes provided by users, and use the network to transfer the segmentations throughout the collection. Our networks deliver not only the basic correspondence required for the transfer, but also propagation paths providing reduced ambiguity and increased reliability. Furthermore, our framework can even be utilized to select the small set of shapes to be segmented by users.

To select the set of shapes to be manually segmented, we stipulate that the best candidates are shapes that are structurally-rich but still similar to other shapes in the collection. A segmentation can then be reliably transferred from more complex to simpler shapes. This requirement is essentially captured by the property related to the extent of influence of a shape, computed in Section 6. Starting from the reachable shapes defined for each shape S_i in the set, we greedily select a set cover spanning the collection to serve as the set of

Variation	Chairs (400)	Vases (300)
K=50	99.17	89.25
Less seeds	99.09	89
K=25	98.74	89.64
K=ALL	99.22	86.52
Clusters	98.65	85.66
Random	95.88	84.79

Table 1: Segmentation accuracy. Shape area percentage that was labeled correctly by our method under several variations as explained in Section 7.

Method	Chairs (400)	Vases (300)
Sidi	80.2	69.9
Kim-Auto	91.2	85.6
Kim-Man	96.9	81.2
FMap	98.1	94.3
SHED	78.8	78.5
PA-H	75.2	82.4
Ours	99.2	89.3

Table 2: Segmentation accuracy. Shape area percentage that was labeled correctly by each method, as explained in Section 7.

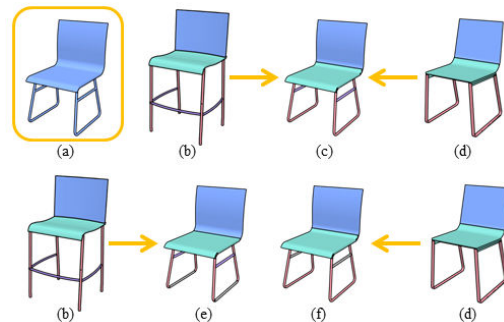


Figure 10: Segmentation propagation from multiple sources. Shape (a) contains parts that correspond to parts in both (b) and (d). By combining the correspondence information given from these two shapes, we obtain a complete and consistent labeling for (a), shown in (c). In contrast, a single source propagation of segmentation from either (b) or (d) yields an incomplete labeling for (a), as shown in (e) and (f), respectively, resulting in unlabeled parts colored in grey.

seed shapes from which to propagate manually-obtained segmentations. We can further narrow down the set of seeds by selecting the top- k most influential shapes in the set cover if necessary.

Given consistent segmentations for the selected seed shapes, we propagate the information through the network. For simplicity, we greedily select, at each step, the pair of shapes S_i and S_j for which $\mathcal{G}_s(i, j) + C_a(i, j)$ is minimal, such that the segmentation for S_i is known, and unknown for S_j . A better approach, however, is to allow propagation of information from multiple sources. In this manner, we reduce the effect of error propagation, and increase the amount of information flowing to a target, possibly covering correspondence holes that may exist in individual connections between a target and its potential sources. See Figure 10 for an example.

To transfer the segmentation from one shape to another, we take an approach similar to the one proposed by Wang et al. [2013].



























































Query	Method	Top-5 retrievals				
	CorrNet					
	HoG					
	Su					
	Su*					
	CorrNet					
	HoG					
	Su					
	Su*					
	CorrNet					
	HoG					
	Su					

Figure 11: Shape retrieval comparison. Top-5 retrievals given by our method (CorrNet), compared to those given by the HoG descriptor and the method of Su et al. [2015] (and Su* for chairs). We note the emphasis on structural similarity of our method as opposed to geometric similarity emphasized by others.

Given a face-level segmentation l_i for shape S_i , we can infer a consistent segmentation l_j for any shape S_j that corresponds with S_i . From the correspondence matrix C^{ij} (see Section 4.3), we create a confidence matrix $A_{f_j \times n_l}$, where f_j and n_l are the number of faces in shape S_j and the number of labels in S_i , respectively. For a face $fc_m^j \in S_j$, we set our confidence in its association with label l as: $A(fc_m^j, l) = \sum_{fc_t^i \in S_i, s.t. \text{label}(fc_t^i)=l} C^{ij}(fc_t^i, fc_m^j) / \sum_{t=1}^{f_i} C^{ij}(fc_t^i, fc_m^j)$, where f_i is the number of faces in S_i . A full segmentation for S_j is then obtained through a graph-cut label optimization with A as the data term, and face adjacency weighted by dihedral angles as the smoothness term.

Figure 9 contains several examples for segmentation propagation. Starting from a consistent face-level labeling provided manually for



































































Query	Condition	Top-5 retrievals				
	Simple					
	Complex					
	Simple					
	Complex					
	Simple					
	Complex					
	Simple					
	Complex					
	Simple					
	Complex					
	Simple					
	Complex					

Figure 12: Directional shape retrieval. Examples of retrieval results for conditional queries, specifying one of two directions: simpler shapes / more complex shapes. The results are returned in the order of their structural similarity to the query.

each source shape on the left, the labeling is propagated through each shape in the path toward the target at the end.

As a quantitative analysis of the efficacy of our method, we compute a set of seed shapes for the chair and vase sets of the COSEG dataset, and propagate the consistent labeling provided with these seed shapes to the entire dataset in the manner described above. The seed selection process recommended 80 (out of 400) seeds for the chair set and 108 (out of 300) seeds for the vase set. From these we select the most influential shapes, and use them as seeds. We compute segmentation accuracy as the percent of correctly labeled surface area, according to the ground-truth.

Table 1 summarizes the accuracy obtained on a number of variations of our approach, designed to evaluate the contribution of our correspondence and similarity estimation, its stability, and the influence of various steps and parameters. Our network of correspon-

dences is constructed for each variation based on the computed similarities, facilitating the propagation of segmentation from the seeds onwards. In our basic configuration, we set $K = 50$, and select the top 15 most recommended seeds for chairs, and 10 for vases. Next, we attempt a similar approach, but decrease the number of seeds to 10 and 7 respectively. In order to examine the influence of K , we then run the basic configuration with $K = 25$, and with $K = \text{ALL}$. In the latter, we essentially ignore the distances given by HoG, and compute a full all-pairs matching across the set. Furthermore, we evaluate the efficacy of our seed selection process by running two additional experiments. First, we perform spectral clustering on the distance matrix \mathcal{G}_s (Section 5), and set the number of clusters to be 15 (chairs) and 10 (vases). From each cluster we select the center-most shape as a seed. Finally, as a baseline, we run our basic configuration 3 times with seeds selected at random, and record the average accuracy among those runs.

These experiments highlight the benefits of our seed selection process, and indicate that our method is stable to changes in parameters. We note the difference in performance between the chair and vase set, explained by the emphasis that is put on structure in our approach. In the vase set, structural differences indeed exist, but geometric ones are more prominent. In contrast, in the chair family, structure is highly varied and provides a strong cue for similarity estimation.

Table 2 summarizes the performance of previous methods on these co-segmentation tasks, as well as that of our basic configuration. We compare the co-segmentation method of Sidi et al. [2011] (Sidi), the template-based method of Kim et al. [2013] with automatic (Kim-Auto) and manual (Kim-Man) template initialization, the functional maps-based method of Huang et al. [2014] (FMap), the projective method of Wang et al. [2013] (PA-H), and the shape edit distance of Kleiman et al. [2015] (SHED). The last two methods, like our method, provide a coupling between similarity and correspondence. Since both methods perform a full matching between shapes, we could not leverage our notion of shape influence for seed selection for propagation. Thus, we used the spectral clustering approach described above. In both experiments we used the distances computed by each respective method to select seeds and determine the propagation routes, similarly to our own scheme.

We note that the method of Huang et al. [2014] (FMap) is more advantageous in its performance on the vase set and its unsupervised nature. In comparison, our method performs slightly better on the chair set and is semi-supervised due to its requirement of segmented seed shapes. However, our method is more scalable to the ever-growing repositories of 3D shapes, as it constructs a sparse network of correspondences within a set spanning thousands of shapes in 1-2 hours. The authors of Huang et al. [2014] state that computing correspondences within a set of 8,401 shapes takes more than a day on a powerful computer with their method.

For an additional comparison to an isometry-based correspondence method [Solomon et al. 2016], please refer to our supplementary material.

Shape retrieval. As shown by Li et al. [2015], the HoG-based Light Field descriptor is a similarity estimator capable of supporting high-performance retrieval tasks. Structural changes naturally induce strong gradients, which are detected by this descriptor. However, this is only done implicitly, therefore the structure is often lost within the global HoG descriptor. As such, it is a valuable tool to locate similar general geometric shape and appearance to a given query, but possibly overlooks important structural differences.

We leverage the structural similarity captured by our matching method for the task of shape retrieval, with an emphasis on structural affinity. To perform the retrieval, we order the nearest neigh-

bors of a query shape according to the matrix \mathcal{C}_s (Section 5). Figure 11 presents the top-5 retrievals obtained by our method for a few selected queries performed on the test set of the SHREC 2016 dataset [Savva et al. 2016], compared to the shapes retrieved by the HoG Light Field descriptor, as well as by the shape retrieval method of Su et al. [2015], by taking the output of the penultimate layer of the model as a comparable feature vector. This method produces poor results since it is geared toward a different task - i.e., multi-class category-based shape retrieval, rather than fine-grained similar shape retrieval. We recognize the unfair disadvantage to the detriment of this method. Thus, to obtain a fair comparison, model instance indices instead of class indices are used as class label supervision to fine-tune the original network of Su et al. [2015]. This is done with the SHREC 2016 chair training set, such that the feature vector serves for fine-grained shape similarity estimation. Similarly to HoG, we observe the tendency of this approach (denoted as Su^* in Figure 11) to focus on the overall geometric similarity rather than local structural details, and attribute it to the aggregated feature vector.

We note that the geometric similarity captured by our matching is a coarse one, composed only of magnitude and depth discrepancies, and is not nearly as fine as the one captured by HoG. As portrayed in the results, our method emphasizes structural rather than stylistic similarities between the shapes. This suggests that combining our approach with a descriptor that measures geometric similarity, such as HoG, can potentially provide a stronger similarity estimation that emphasizes both geometry and topology.

Directional shape retrieval. Another application of the relative structural complexity captured by the matrix \mathcal{C}_d (Section 5), is constrained shape retrieval, where we seek to handle queries requesting similar shapes that are either of a simpler or a more complex structure than the query. For a given query shape S_i , we collect two sets of shapes from the i -th row and column in \mathcal{C}_d . The first set contains any shape S_j for which $\mathcal{C}_d(j, i) \leq \theta_{\text{str}}$ (S_i contains nearly all the regions of S_j) and $\mathcal{C}_d(i, j) \geq \theta_{\text{str}^*}$ (S_j does not contain all the regions of S_i), implying that S_j is of a simpler structure to that of S_i . The second set is formed by reversing the roles, and is therefore populated by shapes that are of a more complex structure. Depending on the type of query (requesting simpler or more complex shapes), we take one of these two sets as the result and present them in order of structural similarity to the query. See Figure 12 for some examples.

Structural collection representation. Exploration of shape collections is an interesting problem in Computer Graphics, combining shape similarity estimation with user interface considerations. In this context, we are interested in the construction of a representation of a collection of shapes that will provide a summary of the available shapes from a structural viewpoint. We note that an important by-product of our matching process, and the structural similarity estimation, is a division of the collection into clusters of structural equivalence. To obtain these clusters, we observe the graph given by the matrix \mathcal{C}_s , and remove all the edges that surpass the strict threshold $0.1 \cdot \theta_{\text{str}}$, thereby eliminating connections that can potentially contradict structural equivalence. Next, we compute the connected components within the graph, yielding clusters of structural affinity. We form a new graph composed of the clusters as nodes, and re-incorporate the removed edges as connectors between clusters. Figure 1 features the minimum spanning tree computed on this cluster graph (using only clusters of substantial size), capturing the structural diversity within the collection and the gradual change in structure given by traversing the paths of the tree. We focus on an example path through a sequence of cluster representatives, and highlight any structural event between successive representatives.

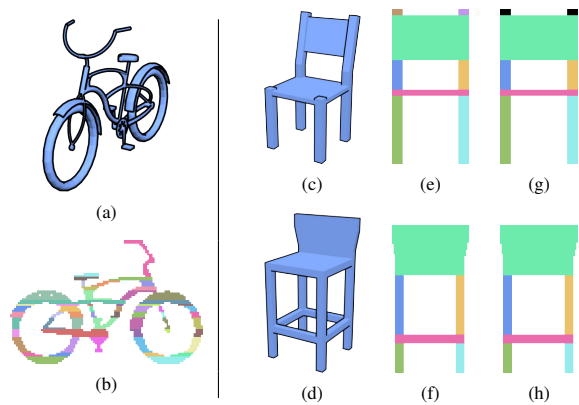


Figure 13: *Limitations of our method. The sensitivity of our decomposition approach to structural changes on the projections may lead to an overly fragmented partitioning that is not meaningful enough for matching, such as the decomposition of the bicycle in (a), shown in (b). Additionally, 2D projections contain limited information pertaining to the volume of the shape extending in the perpendicular direction to the projection plane. Thus, the two projections, taken from the back viewpoint, of chairs (c) and (d), shown in (e) and (f), appear closely similar, leading to the matching shown between (g) and (h), where the seat of (c) is matched to the connecting bar of (d).*

8 Conclusions, limitations and future work

We presented a method to construct a network of correspondences spanning an entire collection of shapes. The method organizes shapes in a graph encoding near-equivalent structure, where some edges signify a change in topology or a significant change of structure between two shapes. Our venture is guided by two meta ideas. The first is the intimate reciprocal relation that exists between similarity and correspondence, implying that a correspondence computed between similar shapes is significantly more reliable. Second, similarity between shapes is generally harder to quantify than structural changes in corresponding shapes. Thus, our method follows a robust pairwise matching process that estimates correspondence and similarity simultaneously, determining the reliability of a connection. Reliable connections can then be leveraged for information propagation within the network.

The notorious difficulty associated with the problem of pairwise shape matching prompted an endeavour to break down the complex 3D problem into many simpler problems, followed by an aggregation of the results. Aiming to emphasize and detect structural differences between shapes as cues for dissimilarity, we rely on these simpler correspondence problems to alleviate the complications involved in the analysis of structure in 3D. These differences, or events, as illustrated in Figure 2, occur when the correspondence is not a one-to-one correspondence. It is important to note that, throughout the analysis process, none of the input shapes undergo any form of explicit segmentation. Instead, the correspondences are established between 2D regions, defined by a local optimization designed to decompose a shape projection into regions with coherent structure. This implicit part-based matching is less sensitive to affine transformations between parts, and greatly contributes to the robustness of connections in the network.

The projective nature of our correspondence estimation is also the main limitation of our method, as it implies sensitivity to articulations. Our method is therefore more suitable for unarticulated shapes, such as man-made objects. Additionally, due to the empha-

sis put on structure in the context of a 2D projection, an intricate shape (e.g. bicycle) may be over-partitioned thereby compromising a meaningful matching (see Figure 13(a)-(b)). Furthermore, despite leveraging depth information given by the projection, some information is naturally lost in the reduction from 3D to 2D. As such, the method is unaware of the extent of shape parts that run orthogonally to the projection plane, potentially leading to an ambiguous correspondence (see Figure 13(c)-(h)). In the future, we are interested in exploring options to address this issue, e.g., by using layered depth images.

Another limitation of our method is related to the conceptual nature of information propagation, which inevitably, is typically accompanied by error propagation. However, the remedy to this is a conservative correspondence and similarity measure. Our method, like other co-analysis methods, assumes that the collection is dense enough, and there are sufficient commonalities among the shapes in the collection. The nature of the propagation, unlike a global co-analysis, allows us to build a connected network that contains shapes of rich variability of geometry and structure. In the future, we would like to make this network dynamic, allowing insertion and removal of shapes, as well as leverage the network as a tool for shape analysis and shape alteration through network traversals.

Finally, to address our set alignment assumption, we suggest a combined scheme where an initial naive alignment sets the ground to an iterative process where both correspondence and alignment are progressively improved based on the outcome of the previous iteration.

For a full implementation of our method please see our project page under <http://www.cs.tau.ac.il/~noafish/cornet>.

Acknowledgements

We thank the anonymous reviewers for their helpful suggestions. This work was supported by a Google Focused Research Award and NSERC (2015-05407).

References

- ALHASHIM, I., XU, K., ZHUANG, Y., CAO, J., SIMARI, P., AND ZHANG, H. 2015. Deformation-driven topology-varying 3D shape correspondence. *ACM Trans. Graph. (SIGGRAPH Asia)* 34, 6, 236:1–13.
- BRONSTEIN, A. M., BRONSTEIN, M. M., GUIBAS, L. J., AND OVSJANIKOV, M. 2011. Shape Google: Geometric words and expressions for invariant shape retrieval. *ACM Trans. Graph.* 30, 1, 1:1–20.
- CHANG, A. X., FUNKHOUSER, T., GUIBAS, L., HANRAHAN, P., HUANG, Q., LI, Z., SAVARESE, S., SAVVA, M., SONG, S., SU, H., XIAO, J., YI, L., AND YU, F. 2015. ShapeNet: An Information-Rich 3D Model Repository. Tech. Rep. arXiv:1512.03012 [cs.GR], Stanford University — Princeton University — Toyota Technological Institute at Chicago.
- CHEN, D.-Y., TIAN, X.-P., SHEN, Y.-T., AND OUHYOUNG, M. 2003. On visual similarity based 3D model retrieval. *Computer Graphics Forum (Eurographics)* 22, 3, 223–232.
- COIFMAN, R. R., AND LAFON, S. 2006. Diffusion maps. *Applied and Computational Harmonic Analysis* 21, 1, 5–30.
- DALAL, N., AND TRIGGS, B. 2005. Histograms of oriented gradients for human detection. In *Proc. IEEE Conf. Comp. Vis. and Pat. Rec. (CVPR)*, vol. 1, 886–893.

- HU, R., FAN, L., AND LIU, L. 2012. Co-segmentation of 3D shapes via subspace clustering. *Computer Graphics Forum (SGP)* 31, 5, 1703–1713.
- HUANG, Q., AND GUIBAS, L. 2013. Consistent shape maps via semidefinite programming. *Computer Graphics Forum (SGP)* 32, 5, 177–186.
- HUANG, Q., KOLTUN, V., AND GUIBAS, L. 2011. Joint shape segmentation with linear programming. *ACM Trans. Graph. (SIGGRAPH Asia)* 30, 6, 125:1–12.
- HUANG, Q., ZHANG, G., GAO, L., HU, S., BUSTCHER, A., AND GUIBAS, L. 2012. An optimization approach for extracting and encoding consistent maps in a shape collection. *ACM Trans. Graph. (SIGGRAPH Asia)* 31, 6, 167:1–11.
- HUANG, Q., WANG, F., AND GUIBAS, L. 2014. Functional map networks for analyzing and browsing large shape collections. *ACM Trans. Graph. (SIGGRAPH)* 33, 4, 36:1–11.
- KAIBEL, V., AND PEINHARDT, M. A. 2006. *On the bottleneck shortest path problem*. Konrad-Zuse-Zentrum für Informationstechnik.
- KALOGERAKIS, E., HERTZMANN, A., AND SINGH, K. 2010. Learning 3D mesh segmentation and labeling. *ACM Trans. Graph. (SIGGRAPH)* 29, 4, 102:1–12.
- KIM, V., LIPMAN, Y., AND FUNKHOUSER, T. 2011. Blended intrinsic maps. *ACM Trans. Graph. (SIGGRAPH)* 30, 4.
- KIM, V. G., LI, W., MITRA, N. J., CHAUDHURI, S., DIVERDI, S., AND FUNKHOUSER, T. 2013. Learning part-based templates from large collections of 3d shapes. *ACM Trans. Graph. (SIGGRAPH)* 32, 4, 70:1–12.
- KLEIMAN, Y., VAN KAICK, O., SORKINE-HORNUNG, O., AND COHEN-OR, D. 2015. SHED: Shape edit distance for fine-grained shape similarity. *ACM Trans. Graph. (SIGGRAPH Asia)* 34, 6, 235:1–11.
- LAGA, H., MORTARA, M., AND SPAGNUOLO, M. 2013. Geometry and context for semantic correspondences and functionality recognition in man-made 3D shapes. *ACM Trans. Graph.* 32, 5, 150:1–16.
- LI, Y., SU, H., QI, C. R., FISH, N., COHEN-OR, D., AND GUIBAS, L. J. 2015. Joint embeddings of shapes and images via CNN image purification. *ACM Trans. Graph. (SIGGRAPH Asia)* 34, 6, 234:1–12.
- LITMAN, R., BRONSTEIN, A., BRONSTEIN, M., AND CASTELLANI, U. 2014. Supervised learning of bag-of-features shape descriptors using sparse coding. *Computer Graphics Forum (SGP)* 33, 5, 127–136.
- NGUYEN, A., BEN-CHEN, M., WELNICKA, K., YE, Y., AND GUIBAS, L. 2011. An optimization approach to improving collections of shape maps. *Computer Graphics Forum* 30, 5, 1481–1491.
- OVSJANIKOV, M., MÉRIGOT, Q., MÉMOLI, F., AND GUIBAS, L. 2010. One point isometric matching with the heat kernel. *Computer Graphics Forum (SGP)* 29, 5, 1555–1564.
- RUSTAMOV, R. M. 2007. Laplace-Beltrami eigenfunctions for deformation invariant shape representation. In *Proc. Symp. Geometry Processing*, 225–233.
- SAVVA, M., YU, F., SU, H., AONO, M., CHEN, B., COHEN-OR, D., DENG, W., SU, H., BAI, S., BAI, X., FISH, N., HAN, J., KALOGERAKIS, E., LEARNED-MILLER, E. G., LI, Y., LIAO, M., MAJI, S., TATSUMA, A., WANG, Y., ZHANG, N., AND ZHOU, Z. 2016. Large-Scale 3D Shape Retrieval from ShapeNet Core55. In *Eurographics Workshop on 3D Object Retrieval*, The Eurographics Association, A. Ferreira, A. Giachetti, and D. Giorgi, Eds.
- SIDI, O., VAN KAICK, O., KLEIMAN, Y., ZHANG, H., AND COHEN-OR, D. 2011. Unsupervised co-segmentation of a set of shapes via descriptor-space spectral clustering. *ACM Trans. Graph. (SIGGRAPH Asia)* 30, 6, 126:1–10.
- SOLOMON, J., PEYRÉ, G., KIM, V., AND SRA, S. 2016. Entropic metric alignment for correspondence problems. *ACM Trans. Graph. (SIGGRAPH)*, to appear.
- SU, H., MAJI, S., KALOGERAKIS, E., AND LEARNED-MILLER, E. 2015. Multi-view convolutional neural networks for 3D shape recognition. In *Proc. ICCV*, 945–953.
- TANGELDER, J. W. H., AND VELTKAMP, R. C. 2008. A survey of content based 3D shape retrieval methods. *Multimedia Tools and Applications* 39, 3, 441–471.
- TENENBAUM, J. B., DE SILVA, V., AND LANGFORD, J. C. 2000. A global geometric framework for nonlinear dimensionality reduction. *Science* 290, 5500, 2319–2323.
- VAN KAICK, O., TAGLIASACCHI, A., SIDI, O., ZHANG, H., COHEN-OR, D., WOLF, L., AND HAMARNEH, G. 2011. Prior knowledge for part correspondence. *Computer Graphics Forum (Eurographics)* 30, 2, 553–562.
- VAN KAICK, O., ZHANG, H., HAMARNEH, G., AND COHEN-OR, D. 2011. A survey on shape correspondence. *Computer Graphics Forum* 30, 6, 1681–1707.
- WANG, Y., ASAFI, S., VAN KAICK, O., ZHANG, H., COHEN-OR, D., AND CHEN, B. 2012. Active co-analysis of a set of shapes. *ACM Trans. Graph. (SIGGRAPH Asia)* 31, 6, 165:1–10.
- WANG, Y., GONG, M., WANG, T., COHEN-OR, D., ZHANG, H., AND CHEN, B. 2013. Projective analysis for 3D shape segmentation. *ACM Trans. Graph. (SIGGRAPH Asia)* 32, 6, 192:1–12.